

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-301555
(43)Date of publication of application : 28.10.1994

(51)Int. CI. G06F 9/46

(21)Application number : 06-022701 (71)Applicant : INTERNATL BUSINESS MACH
CORP <IBM>
(22)Date of filing : 21.02.1994 (72)Inventor : LARRY KEITH ROUX

(30)Priority

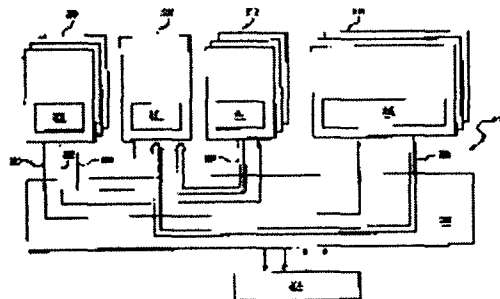
Priority number 93 23666 Priority date 26.02.1993 Priority country US
: : :

(54) SYSTEM FOR PLURAL COEXISTING OPERATING SYSTEM PERSONALITIES ON MICRO-KERNEL

(57)Abstract:

PURPOSE: To make adjusted processing feasible for a plurality of operating systems by selecting one operating system as a main personality and utilizing the other operating systems as auxiliary personalities.

CONSTITUTION: A basic micro-kernel layer 202 offers standard interfaces to a hardware layer 204 and an operating system personality 206. Whenever an application 208 makes a request peculiar to an operating system, an interface 209 converts the request into a service request by means of an auxiliary personality server 214, personality neutral server 212, or main personality server 210. The request is processed in an outward trip shown by the arrows 220, 222, and 224 and the answer is given to the application 208. When required, the answer is adjusted by the main personality server 210.



LEGAL STATUS

[Date of request for examination] 21.02.1994

[Date of sending the examiner's decision 08.10.1996
of rejection]

[Kind of final disposal of application withdrawal
other than the examiner's decision of
rejection or application converted
registration]

[Date of final disposal for application] 05.10.2000

[Patent number]

[Date of registration]

[Number of appeal against examiner's 08-19892
decision of rejection]

[Date of requesting appeal against 29.11.1996
examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁(JP)

(12)公開特許公報(A)

(11)特許出願公開番号

特開平6-301555

(43)公開日 平成6年(1994)10月28日

(51)Int.Cl.⁵

G 0 6 F 9/46

識別記号

3 4 0 A 8120-5B

庁内整理番号

F I

技術表示箇所

審査請求 有 発明の数 9 OL (全 8 頁)

(21)出願番号 特願平6-22701

(22)出願日 平成6年(1994)2月21日

(31)優先権主張番号 0 2 3 6 6 6

(32)優先日 1993年2月26日

(33)優先権主張国 米国(US)

(71)出願人 390009531

インターナショナル・ビジネス・マシーンズ・コーポレーション

INTERNATIONAL BUSINESS MACHINES CORPORATION

アメリカ合衆国10504、ニューヨーク州
アーモンク (番地なし)

(72)発明者 ラリー・キース・ラウクス

アメリカ合衆国78750 テキサス州オースチン
ピックフェア・ドライブ10801

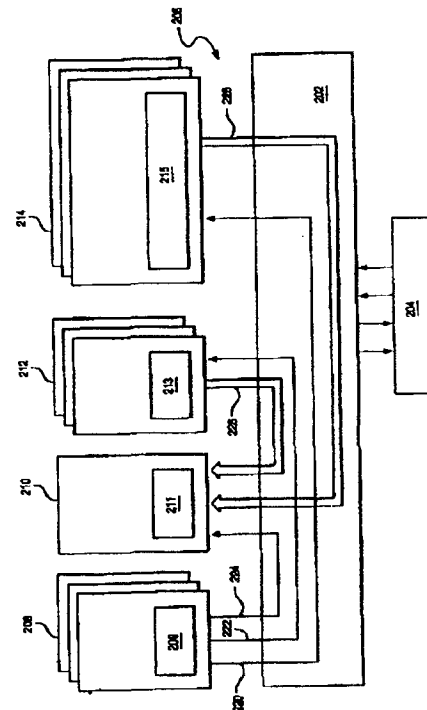
(74)代理人 弁理士 合田 潔 (外2名)

(54)【発明の名称】 マイクロカーネル上の複数の共存オペレーティング・システム・パーソナリティ用のシステム

(57)【要約】

【目的】 ハードウェア・プラットフォーム上で複数のオペレーティング・システム環境を並行に動作するためのシステムおよび方法を提供すること。

【構成】 主パーソナリティ・サーバが基本システム構成および資源を制御するために設けられる。副サーバが、他のオペレーティング・システムのアプリケーションをサポートするために設けられる。主および副オペレーティング・システム・パーソナリティのそれぞれについて2進互換性がサポートされる。パーソナリティ中立サービスが、各オペレーティング・システム環境から抽出され、パーソナリティ中立サーバ中に合体される。プロセス間通信は、定義済みのオブジェクト指向インタフェースおよび手順インタフェースを使って実施される。サーバ応答および競合解消は、主パーソナリティ中立サーバによって処理される。



【特許請求の範囲】

【請求項1】基本カーネル・サービスを実行する手段と、
第1のオペレーティング・システム・プロトコルに従ってオペレーティング・システム・サービスを提供することによりシステム動作全体を制御する主パーソナリティ手段と、
アプリケーション・プログラムからオペレーティング・システム・サービス要求を受け取るアプリケーション・インタフェース手段と、
前記第1のオペレーティング・システム・プロトコルと異なる1つまたは複数のオペレーティング・システム・プロトコルに従ってオペレーティング・システム・サービスを提供する副パーソナリティ手段と、
前記基本カーネル・サービスを実行する前記手段と、前記主パーソナリティ手段と、前記アプリケーション・インタフェース手段と、前記副パーソナリティ手段とをリンクするインタフェース手段とを備える、複数の共存するオペレーティング・システムをサポートするためのシステム。
【請求項2】前記基本カーネル・サービスを実行する前記手段がMac hベースのマイクロカーネルであることを特徴とする、請求項1に記載のシステム。
【請求項3】前記インタフェース手段が、オブジェクト指向メッセージを評価し、それらに作用する手段と、
手順呼出しを評価し、それらに作用する手段とを含むことを特徴とする、請求項1に記載のシステム。
【請求項4】前記主パーソナリティ手段がA I X（米国IBM社の登録商標）動作環境であることを特徴とする、請求項1に記載のシステム。
【請求項5】前記主パーソナリティ手段がOS/2（米国IBM社の登録商標）動作環境であることを特徴とする、請求項1に記載のシステム。
【請求項6】前記副パーソナリティ手段が、OS/2動作環境、DOS動作環境、およびマッキントッシュ動作環境を含むことを特徴とする、請求項4に記載のシステム。
【請求項7】前記副パーソナリティ手段が、A I X動作環境、DOS動作環境、およびマッキントッシュ動作環境を含むことを特徴とする、請求項5に記載のシステム。
【請求項8】それぞれ複数のアプリケーション・タスクのどれにでもアクセスでき、コンピュータ・プロセッサとメモリとを有するプロセッサ・システムと相互作用する、複数のオペレーティング・システム環境を並行して操作する方法であって、第1のオペレーティング・システムを主動作環境プロセスとしてブートするステップと、
複数の副オペレーティング・システム環境を別々のプロ

セスとしてロードするステップと、
前記主動作環境および前記複数の副オペレーティング・システム環境からのメッセージに応答して1つまたは複数のパーソナリティ中立サーバをロードするステップと、
アプリケーションをアプリケーション・プロセスとして実行するステップと、
カーネル・サービスを使用して、プロセス間通信要求およびプロセッサ・システム特有の要求を処理するステップとを備えることを特徴とする方法。
【請求項9】前記カーネル・サービスが米国カーネギー・メロン大学開発のMac hマイクロカーネル・サービスであることを特徴とする、請求項8に記載の方法。
【発明の詳細な説明】
【0001】
【産業上の利用分野】本発明は、全般的に、情報処理システムに関し、詳しくは、オペレーティング・システムに関する。さらに詳細には、本発明は、マイクロカーネル・アーキテクチャを使用して複数の共存オペレーティング・システム・パーソナリティをサポートする方法に関する。
【0002】
【従来の技術】コンピュータ・オペレーティング・システムは、コンピュータ・ハードウェア資源の動作を制御して有用なタスクを実行する。歴史的に見ると、オペレーティング・システムの開発は、ハードウェア製造業者によって、自社の特定のハードウェア・ブランドを制御するオペレーティング・システムを提供するために行われてきた。「開放型」システムの出現によって、オペレーティング・システム開発の焦点が変化した。「開放型」システムは、アプリケーション・プログラム・コードに僅かな変更を加えるだけで複数のハードウェア層、すなわち、ハードウェア・プラットフォーム上で動作することが期待される。
【0003】コンピュータ・ハードウェアの変化する速度がますます速くなっており、オペレーティング・システム開発もその影響を受けている。ハードウェアが絶えず変化するので、既存のオペレーティング・システムを新規のハードウェア・プラットフォームに適合させるのに必要な作業およびコストが増大している。
【0004】このような傾向のために、コンピュータ業界はオペレーティング・システム開発の代替方法を模索し始めた。カーネギー・メロン大学は、「マイクロカーネル」アーキテクチャを開発するプロジェクトに着手した。このプロジェクトで、Mac hマイクロカーネルが開発され、多数のハードウェア・ベンダに提供され、これらのベンダによって適合されている。マイクロカーネル手法では、基本ハードウェア資源の制御をオペレーティング・システム環境に固有のインタフェースから分離しようと試みている。ハードウェア制御プロセスは、機

10

20

30

40

50

能の最小(「マイクロ」)カーネルにグループ化される。このコードを新規のハードウェア・アーキテクチャに適合させるのは、ハードウェア依存性が全体に散在している、より大規模なオペレーティング・システムを適合させるよりもはるかに簡単である。さらに、マイクロカーネルへのインタフェースは不変のままなので、新規のハードウェアに適合させるためにオペレーティング・システムの高次層を変更しなくて済む。

【0005】マイクロカーネル手法は、オペレーティング・システムを新規のハードウェア・プラットフォームに迅速に適合させ、多数のオペレーティング・システムを既存のプラットフォームに迅速に適合できるようにする方法として提案された。特定のハードウェア・アーキテクチャ用の各システム・プロダクトに単一または複数のオペレーティング・システム・パーソナリティ(personality)を与えることが提案されている。より詳しい背景については、リチャード・ラシード(Richard Rashid)著"A Catalyst for Open System"(Datamation, 1989年5月15日, pp. 32-33)を参照されたい。パーソナリティとは、OSカーネル(たとえば、マイクロカーネル)上で動作するDOS、Windows、AIX等のOSサービスのことである。

【0006】カーネギー・メロン大学(CMU)のMachマイクロカーネルの開発者は、単一のマイクロカーネル上で走行する複数のオペレーティング・システム・パーソナリティをサポートすることを提案している。この手法は、"Generalized Emulation Services for Mach 3.0 - Overview, Experience and Current Status"(USENIX Association - Mach Symposium Proceedings, 1991年11月, pp. 13-26)で論じられている。

【0007】前記の論文では、エミュレーションによって複数の対象のオペレーティング・システムを提供することが提案されている。この場合、エミュレーション・ライブラリを、エミュレートされた各プロセスと関連付けて、そのプロセスに必要なオペレーティング・システム環境のエミュレーションが提供される。提案されたシステムでは、専用オペレーティング・システム・エミュレーションを、ファイル管理、端末通信、プロセス管理などの共通タスクを処理する汎用サーバと組み合わせている。汎用サーバは、別個のプロセスとして実行され、Machメッセージを介してエミュレーション・ライブラリと通信する。

【0008】しかし、このCMUの手法は、性能を大幅に低下させずに複数のオペレーティング・システムを同時にサポートできるほど一般化されていない。CMUによって提案されたエミュレーション・ライブラリは、アプリケーション・プロセスのアドレス空間内で走行する。オペレーティング・システムの複雑さが多くはエミュレーション・ライブラリに組み込まれるため、ライ

ブラリが大規模かつ複雑になる。エミュレートされたライブラリを使用する各アプリケーションは、そのライブラリをそれ自体のアドレス空間にロードする必要がある。したがって、メモリ・オーバーヘッドとシステム資源消費量が重大な問題となる。最後に、サポートされる各オペレーティング・システムがエミュレートされるので、オペレーティング・システムはすべて、エミュレーション環境によって性能が低下する。

【0009】複数のオペレーティング・システム・パーソナリティをサポートできる操作環境を高い費用効果で開発するには、オペレーティング・システム間の共通要素を抽出して合体し、制限された数のプロセスにする必要がある。また、システム・ユーザが全体的システム機能を制御する単一の主パーソナリティすなわち、ドメイン(domain)パーソナリティ、を選択できるようにすると共に、必要に応じて他のオペレーティング・システムを使用することが好ましい。

【0010】

【発明が解決しようとする課題】本発明が対象とする技術的問題は、ハードウェア・システム上で複数の並行なオペレーティング・システムを効率的にサポートするためのシステムおよび方法を開発することである。第2の問題は、必然的に複数のシステムによって共用されるハードウェア資源と通信するための効果的な手段を提供することである。

【0011】本発明は、複数の並行なオペレーティング・システム・パーソナリティをサポートするためのシステムおよび方法を実施することによりこれらの問題を解決することを対象とする。

【0012】本発明は、オペレーティング・システム・パーソナリティ間で資源を調整し、パーソナリティ間で効果的に通信を行うための構成要素を有するシステムを提供する。効果的な通信の方法も開示する。

【0013】したがって、本発明の目的は、アプリケーション・プロセスに対して透過的な並行オペレーティング・システム・サポートを提供することである。

【0014】本発明の他の目的は、主パーソナリティの性能を低下させずに、複数の並行なオペレーティング・システムのサポートを提供することである。

【0015】本発明の他の目的は、以前からのオペレーティング・システム・アプリケーションを保存するシステムを提供することである。

【0016】

【課題を解決するための手段】本発明は、基本カーネル・サービスを実行する手段と、第1のオペレーティング・システム・プロトコルに従ってオペレーティング・システム・サービスを提供することによりシステム動作全体を制御する主パーソナリティ手段と、アプリケーション・プログラムからオペレーティング・システム・サービス要求を受け取るアプリケーション・インタフェース

手段と、第1のオペレーティング・システム・プロトコルと異なる1つまたは複数のオペレーティング・システム・プロトコルに従ってオペレーティング・システム・サービスを提供する副パーソナリティ手段と、基本カーネル・サービスを実行する手段、主パーソナリティ手段、アプリケーション・インタフェース手段、および副パーソナリティ手段をリンクするインタフェース手段とを有する、共存する複数のオペレーティング・システムをサポートするシステムを対象とする。

【0017】図面では、同じ符号は、本発明の同じ部品を表す。

【0018】

【実施例】マイクロカーネルは、特定のハードウェア・アーキテクチャのハードウェア制御プロセスを、容易に修正できる単一のマイクロカーネルにカプセル化するように設計されている。この手法は、多数の異なるハードウェア・アーキテクチャでマイクロカーネル・インタフェースを使用できるように設計されている。図1に、マイクロカーネル・ベースのオペレーティング・システムが動作するハードウェア構成の例を示す。基本構成要素には、1つまたは複数の処理装置またはCPU102、ハード・ディスクまたはその他の永久的な記憶装置104、ランダム・アクセス・メモリ(RAM)106、アダプタ108を介するネットワーク通信サポート、ならびに入出力制御装置110を介する表示装置112、キーボード114、およびポインティング装置116への入出力サポートが含まれる。これらの構成要素は、システム・バス103を介して通信する。

【0019】図1の汎用システムは、IBM PS/2コンピュータやIBM RISCシステム/6000コンピュータ(PS/2およびRISC/6000はIBMの商標)など多数の市販のシステムで実施される。本発明の並行オペレーティング・システム・サポートは、これらのまたはマイクロカーネルを実行するように適合された新規のどんなハードウェア・アーキテクチャ上でも機能するように設計されている。

【0020】本発明のシステムを図2に概略的に示す。基本マイクロカーネル層は202に示されている。マイクロカーネルは、ハードウェア層204への標準インタフェースと、オペレーティング・システム・パーソナリティ206への標準インタフェースを提供する。並行オペレーティング・システム・サポートを実施する本発明の主な構成要素は、208、210、212、214に示されている。これらの構成要素は、定義されたインタフェース209、211、213、215によってリンクされる。

【0021】好ましい実施例のシステムは、ハードウェア・システムに対する全体的制御を提供する主パーソナリティ・サーバ210を有する。主パーソナリティ・サーバ210は、ハードウェア・システムの初期ブート時

に制御を与えられ、基本オペレーティング・システム構成要素を確立する。主パーソナリティ・サーバ210はまた、通信アダプタや表示装置などの共用システム資源へのアクセスを調整する。主パーソナリティ・サーバ210は、インタフェース211を介して他のプロセスと通信する。

【0022】インタフェース211は、オブジェクト指向インタフェースならびに標準手順インタフェースを含む。オブジェクト指向インタフェースは、主パーソナリティから送られるオブジェクト・メッセージに応答し、あるいは主パーソナリティに応答する。手順インタフェースは、何らかのコマンドまたはルーチン呼出しと、パラメータを受け入れる従来型の実用アプリケーション・プログラミング・インタフェース(API)である。インタフェース211は、Macintoshマイクロカーネル・メッセージ・サービスを介して他のプロセスと通信する。

【0023】パーソナリティ中立(neutral)サービス212は、オペレーティング・システムの特徴を有する必要がない一般的なタスクをサポートするために提供される。このようなサービスの例として、ファイル・システム、通信移送サービス、および分散システム・サービスがある。これらのパーソナリティ中立サービス212は、どのオペレーティング・システム・パーソナリティからのプロセス要求でも受け入れ、必要なサービスを提供する。これらのサービスの調整は、オブジェクト指向インタフェースおよび手順インタフェース213を介して行われる。

【0024】その他のオペレーティング・システム・パーソナリティは、副(sub-dominant)パーソナリティ・サーバ214として実施される。これらのインタフェースはそれぞれ、インタフェース215を介してマイクロカーネルおよび他のサーバと相互接続される。その他のパーソナリティは、主パーソナリティ環境と異なるパーソナリティ環境用のオペレーティング・システム環境を提供する。また、パーソナリティ特有のサービス216をさらに分離することもできる。

【0025】システムまたはユーザが実行するアプリケーションが208に示されている。アプリケーションは、主パーソナリティやマイクロカーネルを知らずに実行される。各アプリケーションはそれぞれ、その対象のオペレーティング・システムがコンピュータ・ハードウェアの完全な制御下にある場合と同様に実行される。

【0026】アプリケーションの処理は、以下のように行われる。アプリケーションが、アプリケーション・タスク領域で実行を開始する。アプリケーションがオペレーティング・システム特有の要求を行うたびに、その要求が、オブジェクト指向/手順フレームワーク209により、マイクロカーネル、副パーソナリティ・サーバ214、パーソナリティ中立サーバ、または主パーソナリティ・サーバ210によるサービス要求に変換される。

これらの要求は、図2に矢印220、222、224で示されている。アプリケーション要求が処理され、アプリケーションに応答が直接届く。必要に応じて、応答が主パーソナリティ・サーバ210によって調整されることもある(矢印226、228)。これによって、たとえば画面への書込みが制御できる。

【0027】プロセス間通信のさらに詳細な流れを図3に示す。図3は、それぞれアプリケーション・タスク、パーソナリティ中立サーバ、および主パーソナリティ・サーバのアドレス空間を表す、3つのセクションに分割

されている。これら3つのセクションは、例として使用したものすぎない。同様な通信経路が他のサーバにも使用される。

【0028】アプリケーション・アドレス空間310で走行するアプリケーション・プログラムが、表示装置画面上のウィンドウにデータを書き込もうとすることがある。この場合、アプリケーションに知られているオペレーティング・システム・プロトコルに従って要求が行われる。書込み要求がインタフェースによってインタセプトされ、処理の決定が下される。インタフェースはま

ず、表示装置ページがアプリケーション・アドレス空間メモリにあるかどうか、およびタスクがそのページにアクセスできるかどうかを検査する。この2つの条件が整っている場合、インタフェースは表示装置に直接書き込むことができる。整っていない場合は、メモリ・ページ不在が発生する。

【0029】メモリ・ページ不在が発生すると、パーソナリティ中立サービス・アドレス空間312の表示装置ドライバにメッセージが送られる。装置ドライバは、必要なページをアプリケーション・メモリに割り当てよう

と試みる。装置ドライバはまず、要求側プロセスにページへのアクセスを許可できるか、ページへの要求側プロセスのアクセスを拒絶すべきか、要求側プロセスはページを待つべきか、あるいは決定を主パーソナリティ・サーバに委せるべきかを検査する。装置ドライバは、状況を知ると、主サーバにメッセージを送らずに、ただちに要求を許可、拒絶、または遅延することができる。装置ドライバは、アプリケーション・プロセスに直接応答する。

【0030】決定が必要な場合、主パーソナリティ・サーバ・アドレス空間314にメッセージが送られる。主パーソナリティは、アクセスを許可または拒絶するための規則を適用し、装置ドライバに応答する。次に、装置ドライバがアプリケーションに応答する。

【0031】アプリケーションが主パーソナリティ以外のオペレーティング・システム環境を使用しているときも類似のプロセスが使用される。その場合、インタフェース論理で処理できないサービス要求は、処置のために、副パーソナリティ・サーバ・タスクに送られる。副パーソナリティ・サーバ214は、システムの方針を適

用する必要があるときは必ず、主パーソナリティ・サーバ210にメッセージを送らなければならない。主パーソナリティ・サーバ210は、すべての資源競合を解消する。

【0032】本発明のシステムは、動作に際して、並行に動作するオペレーティング・システム環境をサポートするための余計なオーバーヘッドをほとんど必要としない。単一のオペレーティング・システムまたはMac h単一パーソナリティ実施態様では、通常、アプリケーションからシステム・レベルまたはカーネルへのサービス要求がある。Mac hは、メッセージを直接処理し、あるいはサーバに処理させるために経路指定することができるが、Mac hアーキテクチャはこのトランザクションにオーバーヘッドをほとんど増加させない。好ましい実施例のメッセージ・ベースの実施態様は、最小メッセージ・トラフィックのモデルに準拠する。主パーソナリティを使用するアプリケーションは、単一オペレーティング・システム実施態様で得られるのと同じ応答を認識ようになる。大部分の副パーソナリティ要求は、同様に迅速に処理される。資源競合例外だけが、主パーソナリティ・サーバ210の検査によってメッセージ・トラフィックが増加する。

【0033】したがって、本発明は、複数のオペレーティング・システム用の調整のとれた処理を実行するという目的を達成し、しかも最小のオーバーヘッドでそれを達成する。

【0034】主パーソナリティの選択は、システム管理者またはシステム設置者が行う。利用可能な各オペレーティング・システムは、主形式および副形式で供給される。管理者は、どのオペレーティング・システムを主パーソナリティとするかを選択し、他のオペレーティング・システムはすべて副パーソナリティとして提供される。副パーソナリティは主サーバのスレーブではないことに留意されたい。副パーソナリティは完全なオペレーティング・システム環境を表す。ただし、ある種の機能は主パーソナリティに取っておかれる。

【0035】以下のとおり発明を開示する。

【0036】(1) 基本カーネル・サービスを実行する手段と、第1のオペレーティング・システム・プロトコルに従ってオペレーティング・システム・サービスを提供することによりシステム動作全体を制御する主パーソナリティ手段と、アプリケーション・プログラムからオペレーティング・システム・サービス要求を受け取るアプリケーション・インタフェース手段と、前記第1のオペレーティング・システム・プロトコルと異なる1つまたは複数のオペレーティング・システム・プロトコルに従ってオペレーティング・システム・サービスを提供する副パーソナリティ手段と、前記基本カーネル・サービスを実行する前記手段と、前記主パーソナリティ手段と、前記アプリケーション・インタフェース手段と、前

記副パーソナリティ手段とをリンクするインタフェース手段とを備える、複数の共存するオペレーティング・システムをサポートするためのシステム。

【0037】(2) 前記基本カーネル・サービスを実行する前記手段がMa c hベースのマイクロカーネルであることを特徴とする、(1)に記載のシステム。

【0038】(3) 前記インタフェース手段が、オブジェクト指向メッセージを評価し、それらに作用する手段と、手順呼出しを評価し、それらに作用する手段とを含むことを特徴とする、(1)に記載のシステム。

【0039】(4) 前記主パーソナリティ手段がA I X(米国I B M社の登録商標)動作環境であることを特徴とする、(1)に記載のシステム。

【0040】(5) 前記主パーソナリティ手段がO S / 2(米国I B M社の登録商標)動作環境であることを特徴とする、(1)に記載のシステム。

【0041】(6) 前記副パーソナリティ手段が、O S / 2動作環境、D O S動作環境、およびマッキントッシュ動作環境を含むことを特徴とする、(4)に記載のシステム。

【0042】(7) 前記副パーソナリティ手段が、A I X動作環境、D O S動作環境、およびマッキントッシュ動作環境を含むことを特徴とする、(5)に記載のシステム。

【0043】(8) それぞれ複数のアプリケーション・タスクのどれにでもアクセスでき、コンピュータ・プロセッサとメモリとを有するプロセッサ・システムと相互作用する、複数のオペレーティング・システム環境を並行して操作する方法であって、第1のオペレーティング・システムを主動作環境プロセスとしてブートするステップと、複数の副オペレーティング・システム環境を別々のプロセスとしてロードするステップと、前記主動作環境および前記複数の副オペレーティング・システム環境からのメッセージに応答して1つまたは複数のパーソナリティ中立サーバをロードするステップと、アプリケーションをアプリケーション・プロセスとして実行するステップと、カーネル・サービスを使用して、プロセス間通信要求およびプロセッサ・システム特有の要求を処理するステップとを備えることを特徴とする方法。

【0044】(9) 前記カーネル・サービスが米国カーネギー・メロン大学開発のMa c hマイクロカーネル・サービスであることを特徴とする、(8)に記載の方法。

【0045】(10) 複数のオペレーティング・システムを並行に実行する、コンピュータ・プログラム論理がその上に記録されたコンピュータ読取り可能媒体を有するコンピュータ・プログラム製品であって、基本カーネル・サービスを実行する手段と、第1のオペレーティング・システム・プロトコルに従ってオペレーティング・システム・サービスを提供することによりシステム動作

全体を制御する主パーソナリティ手段と、アプリケーション・プログラムからオペレーティング・システム・サービス要求を受け取るアプリケーション・インタフェース手段と、前記第1のオペレーティング・システム・プロトコルと異なる1つまたは複数のオペレーティング・システム・プロトコルに従ってオペレーティング・システム・サービスを提供する副パーソナリティ手段と、基本カーネル・サービスを実行する前記手段と、前記主パーソナリティ手段と、前記アプリケーション・インタフェース手段と、前記副パーソナリティ手段とをリンクするインタフェース手段とを備えることを特徴とする、コンピュータ・プログラム。

【0046】(11) 前記基本カーネル・サービスを実行する前記手段がMa c hベースのマイクロカーネルであることを特徴とする、(10)に記載のコンピュータ・プログラム。

【0047】(12) 前記インタフェース手段が、オブジェクト指向メッセージを評価し、それらに作用する手段と、手順呼出しを評価し、それらに作用する手段とを含むことを特徴とする、(10)に記載のコンピュータ・プログラム。

【0048】(13) 前記主パーソナリティ手段がA I X動作環境であることを特徴とする、(10)に記載のコンピュータ・プログラム。

【0049】(14) 前記主パーソナリティ手段がO S / 2動作環境であることを特徴とする、(10)に記載のコンピュータ・プログラム。

【0050】(15) 前記副パーソナリティ手段が、O S / 2動作環境、D O S動作環境、およびマッキントッシュ動作環境を含むことを特徴とする、(13)に記載のコンピュータ・プログラム。

【0051】(16) 前記副パーソナリティ手段が、A I X動作環境、D O S動作環境、およびマッキントッシュ動作環境を含むことを特徴とする、(14)に記載のコンピュータ・プログラム。

【0052】

【発明の効果】複数の並行オペレーティング・システム・パーソナリティを提供することにより、システム・ユーザが、D O S、Windows、O S / 2、A I X、またはその他のオペレーティング・システム用に書かれた以前からのアプリケーションを引き続き使用しながら、新しいオペレーティング・システム・モデルに移行することが可能になる。またこの構造により、開発者が新規のオペレーティング・システムを開発し、新規のハードウェアに適合させることがさらに容易になる。特に、既存のオペレーティング・システム・サービスへの投資が保護される。

【図面の簡単な説明】

【図1】本発明の好ましい実施例を実施することのできるコンピュータ・システムのブロック図である。

11

12

【図2】本発明の好ましい実施例によるシステムのブロック図である。

【図3】本発明の好ましい実施例のアプリケーション使用に含まれる工程ステップを示すフロー・チャートである。

【符号の説明】

102 CPU

103 システム・バス

104 記憶装置

106 ランダム・アクセス・メモリ (RAM)

108 アダプタ

110 入出力制御装置

112 表示装置

114 キーボード

* 116 ポインティング装置

202 基本マイクロカーネル層

204 ハードウェア層

206 オペレーティング・システム・パーソナリティ

209 インタフェース

210 主パーソナリティ・サーバ

211 インタフェース

212 パーソナリティ中立サービス

213 インタフェース

10 214 副パーソナリティ・サーバ

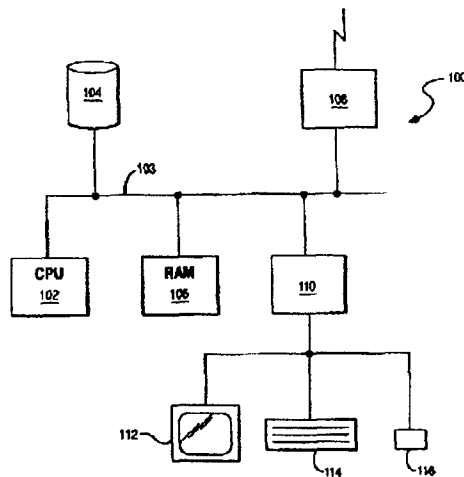
215 インタフェース

310 アプリケーション・アドレス空間

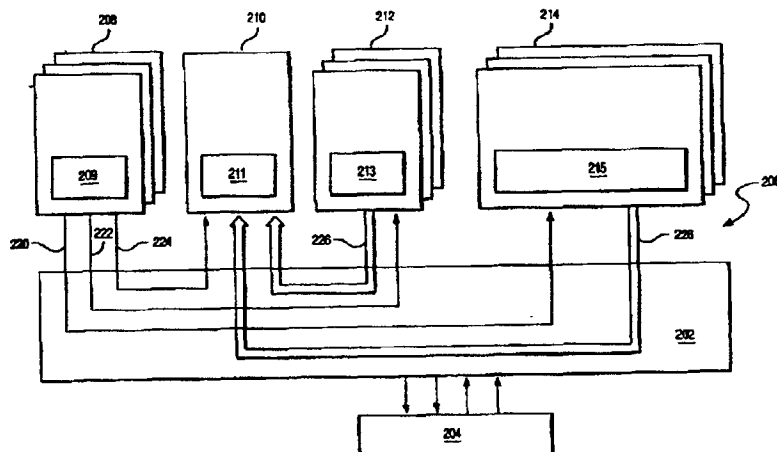
312 パーソナリティ中立サービス・アドレス空間

* 314 主パーソナリティ・アドレス空間

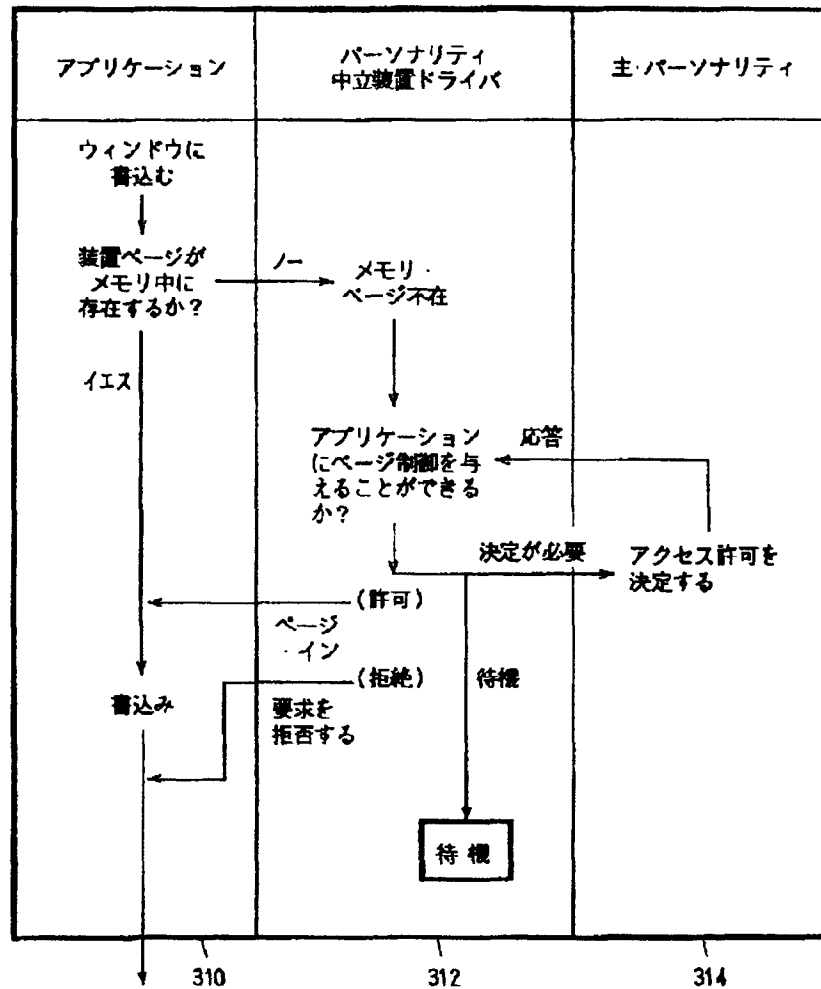
【図1】



【図2】



【図3】



PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-301555

(43)Date of publication of application : 28.10.1994

(51)Int.Cl.

G06F 9/46

(21)Application number : 06-022701

(71)Applicant : INTERNATL BUSINESS MACH
CORP <IBM>

(22)Date of filing : 21.02.1994

(72)Inventor : LARRY KEITH ROUX

(30)Priority

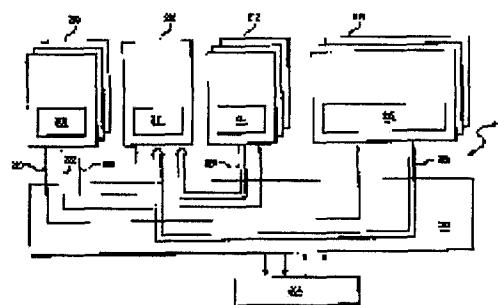
Priority number : 93 23666 Priority date : 26.02.1993 Priority country : US

(54) SYSTEM FOR PLURAL COEXISTING OPERATING SYSTEM PERSONALITIES ON MICRO-KERNEL

(57)Abstract:

PURPOSE: To make adjusted processing feasible for a plurality of operating systems by selecting one operating system as a main personality and utilizing the other operating systems as auxiliary personalities.

CONSTITUTION: A basic micro-kernel layer 202 offers standard interfaces to a hardware layer 204 and an operating system personality 206. Whenever an application 208 makes a request peculiar to an operating system, an interface 209 converts the request into a service request by means of an auxiliary personality server 214, personality neutral server 212, or main personality server 210. The request is processed in an outward trip shown by the arrows 220, 222, and 224 and the answer is given to the application 208. When required, the answer is adjusted by the main personality server 210.



LEGAL STATUS

[Date of request for examination] 21.02.1994

[Date of sending the examiner's decision of rejection] 08.10.1996

[Kind of final disposal of application other than withdrawal the examiner's decision of rejection or application converted registration]

[Date of final disposal for application] 05.10.2000

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection] 08-019892

[Date of requesting appeal against examiner's decision of rejection] 29.11.1996

[Date of extinction of right]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] A means to perform basic kernel service, and a main personality means to control the whole system behavior by offering operating system service according to the 1st operating system protocol, An application interface means to receive an operating system service request from an application program, A vice-personality means to offer operating system service according to one or more different operating system protocols from said 1st operating system protocol, Said means to perform said basic kernel service, and said main personality means, The system equipped with an interface means to link said application interface means and said vice-personality means for supporting the operating system with which plurality lives together.

[Claim 2] The system according to claim 1 characterized by said means to perform said basic kernel service being the microkernel of the Mach base.

[Claim 3] The system according to claim 1 by which said interface means evaluates an object-oriented message, and is characterized by including a means to act on them, and a means to evaluate a procedure call and to act on them.

[Claim 4] The system according to claim 1 characterized by said main personality means being AIX (trademark of U.S. IBM) operating environment.

[Claim 5] The system according to claim 1 characterized by said main personality means being OS/2 (trademark of U.S. IBM) operating environment.

[Claim 6] The system according to claim 4 by which said vice-personality means is characterized by including OS/2 operating environment, DOS operating environment, and the Macintosh operating environment.

[Claim 7] The system according to claim 5 by which said vice-personality means is characterized by including AIX operating environment, DOS operating environment, and the Macintosh operating environment.

[Claim 8] Which of two or more application tasks can be accessed, respectively. Interact with the processor system which has a computer processor and memory. The step which is the approach of operating two or more operating system environments in parallel, and boots the 1st operating system as a main movement environmental process, The step which loads two or more suboperating system environments as a separate process, The step which answers a message from said main movement environment and said two or more suboperating system environments, and loads one or more personality neutral servers, The approach characterized by having the step which performs application as an application process, and the step which uses kernel service and processes an interprocess communication demand and a demand peculiar to a processor system.

[Claim 9] The approach according to claim 8 characterized by said kernel service being Mach microkernel service of the U.S. Carnegie Mellon University development.

[Translation done.]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.**** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Industrial Application] This invention relates to an operating system in detail about information processing system generally. Furthermore, this invention relates to the approach of supporting two or more coexistence operating system personalities using microkernel architecture at a detail.

[0002]

[Description of the Prior Art] A computer operating system controls actuation of computer hardware resources, and performs a useful task. If it sees historically, development of an operating system will have been performed by the hardware manufacturer in order to offer the operating system which controls the specific hardware brand of its company. The focus of operating system development changed with the advent of an "open sand mold" system. Operating on two or more hardware layers only by an "open sand mold" system adding slight modification to an application program code, i.e., a hardware plat form, is expected.

[0003] The rate from which computer hardware changes is still quicker, and operating system development is also influenced [the]. Since hardware changes continuously, an activity and cost required to fit the existing operating system to a new hardware plat form are increasing.

[0004] Computer business began to grope for the alternate method of operating system development for such an inclination. Carnegie Mellon University started the project which develops "microkernel" architecture. In this project, a Mach microkernel is developed, and it is provided for many hardware vendors, and conforms by these vendors. By the microkernel technique, it has tried to separate control of basic hardware resources into an operating system environment from the interface of a proper. Grouping of the hardware-control process is carried out to the minimum ("micro") kernel of a function. It is far easy to fit this code to new hardware architecture rather than it fits the larger-scale operating system with which hardware dependence nature is scattered in the whole. Furthermore, since the interface to a microkernel is still eternal, in order to make new hardware suit, it is not necessary to change the high order layer of an operating system.

[0005] The microkernel technique was proposed as an approach of fitting an operating system to a new hardware plat form promptly, and enabling it to suit the existing plat form promptly in many operating systems. Giving a single or two or more operating system personalities (personality) to each system product for specific hardware architecture is proposed. Please refer to "A Catalyst for Open System" written by Richard Rashid (Richard Rashid) (Datamation, May 15, 1989, pp.32-33) about a more detailed background. Personalities are OS services, such as DOS, Windows, AIX, etc. which operate on OS kernel (for example, microkernel).

[0006] The developer of the Mach microkernel of Carnegie Mellon University (CMU) has proposed supporting two or more operating system personalities who run on a single microkernel. This technique is discussed by "Generalized Emulation Services for Mach 3.0-Overview and Experience and Current Status" (USENIX Association-Mach Symposium Proceedings, November, 1991, pp.13-26).

[0007] In the aforementioned paper, offering the operating system of two or more objects by the emulation is proposed. In this case, an emulation library is related with each emulated process, and the emulation of a required operating system environment is offered in that process. The specialized-operating-system emulation is combined with the general-purpose server which processes common tasks, such as file management, a terminal communication link, and process control, in the proposed system. A general-purpose server is performed as a separate process, and communicates with an emulation library through a Mach message.

[0008] However, this technique of CMU is not generalized, so that two or more operating systems can be supported simultaneously, without reducing the engine performance substantially. The emulation library proposed by CMU runs in the address space of an application process. Since many are included in an emulation library, the complexity of an operating system becomes on a large scale [a library] and complicated. Each application which uses the emulated library needs to load the library to the address space of itself. Therefore, a memory overhead and system-resource consumption pose a serious problem. Since each operating system supported is emulated at the end, as for all operating systems, the engine performance falls according to an emulation environment.

[0009] In order to develop the operating environment which can support two or more operating system personalities by high cost effectiveness, it extracts, and it is necessary to coalesce and to make the common element between operating systems into a restricted number of processes. Moreover, while a system user enables it to choose the single main personality, i.e., a domain (domain) personality, who controls an overall system function, it is desirable that other operating systems can be used if needed.

[0010]

[Problem(s) to be Solved by the Invention] The target technical issue [this invention] is developing the system and approach for supporting two or more parallel operating systems efficiently on a hardware system. The 2nd problem is offering the effective means for communicating with the hardware resources inevitably shared by two or more systems.

[0011] This invention is aimed at solving these problems by enforcing the system and approach for supporting two or more parallel operating system personalities.

[0012] This invention adjusts a resource among operating system personalities, and offers among personalities the system which has a component for communicating effectively. The approach of an effective communication link is also indicated.

[0013] Therefore, a one division target of this invention is offering a transparent concurrency operating system support to an application process.

[0014] Other objects of this invention are offering the support of two or more parallel operating systems, without reducing the main personality's engine performance.

[0015] Other objects of this invention are offering the system which saves the operating system application from before.

[0016]

[Means for Solving the Problem] A means by which this invention performs basic kernel service, and a main personality means to control the whole system behavior by offering operating system service according to the 1st operating system protocol, An application interface means to receive an operating system service request from an application program, A vice-personality means to offer operating system service according to one or more different operating system protocols from the 1st operating system protocol, A means to perform basic kernel service, the main personality means, an application interface means, And it is aimed at the system which has an interface means to link a vice-personality means and which supports two or more operating systems which live together.

[0017] The same sign expresses the same components of this invention with a drawing.

[0018]

[Example] The microkernel is designed so that the hardware-control process of specific hardware architecture may be encapsulated to the single microkernel which can correct easily. This technique is

designed so that a microkernel interface can be used by the hardware architecture from which a large number differ. The example of the hardware configuration to which the operating system of the microkernel base operates to drawing 1 is shown. The I/O support to the network communication support through one or more processors or CPU102, a hard disk or the other eternal storage 104, random access memory (RAM) 106, and an adapter 108 and the indicating equipment 112 through an input/output control unit 110, a keyboard 114, and pointing equipment 116 is included in a basic component. These components communicate through a system bus 103.

[0019] The general-purpose system of drawing 1 is IBM. A PS/2 computer and IBM It carries out by systems of much marketing, such as a RISC system / 6000 computer (PS/2, and RISC/6000 are the trademark of IBM). The concurrency operating system support of this invention is designed so that it may function on any new hardware architecture which is these, or suited so that a microkernel might be performed.

[0020] The system of this invention is roughly shown in drawing 2 . The basic microkernel layer is shown in 202. A microkernel offers the standard interface to the hardware layer 204, and the standard interface to the operating system personality 206. The main components of this invention which carries out a concurrency operating system support are shown in 208, 210, 212, and 214. These components are linked with the defined interfaces 209, 211, 213, and 215.

[0021] The system of a desirable example has the main personality server 210 which offers the overall control to a hardware system. The main personality server 210 can give control at the time of initial boot of a hardware system, and establishes a basic-operating-system component. The main personality server 210 adjusts access to shared bus system resources, such as a communication adapter and a display, again. The main personality server 210 communicates with other processes through an interface 211.

[0022] An interface 211 includes an object-oriented interface and a standardized procedure interface. An object-oriented interface answers the object message sent by the main personality, or answers the main personality. Procedural interfaces are a certain command or a routine call, and an application programming interface (API) of the conventional type which accepts a parameter. An interface 211 communicates with other processes through Mach microkernel messaging service.

[0023] The personality neutral (neutral) service 212 is offered in order to support the general task which does not need to have the feature of an operating system. As an example of such service, there are a file system, communication link migration service, and distributed-system service. These personality neutral services 212 also accept the process demand from which operating system personality, and supply required service. Adjustment of these services is performed through an object-oriented interface and a procedural interface 213.

[0024] Other operating system personalities are carried out as a ** (sub-dominant) personality server 214. These interfaces interconnect with a microkernel and other servers through an interface 215, respectively. Other personalities offer a different operating system environment for personality environments from the main personality environment. Moreover, the service 216 peculiar to a personality is also further separable.

[0025] The application which a system or a user performs is shown in 208. Application is performed by not knowing in the main personality or a microkernel. Each application is performed like the case where the operating system of the object is under perfect control of computer hardware, respectively.

[0026] Processing of application is performed as follows. Application starts activation in an application task field. The demand is changed into the service request by the microkernel, the vice-personality server 214, the personality neutral server, or the main personality server 210 whenever it performs the demand with application peculiar to an operating system according to object-oriented / procedure framework 209. These demands are shown to drawing 2 by arrow heads 220, 222, and 224. An application demand is processed and a response reaches application directly. A response may be adjusted by the main personality server 210 if needed (arrow heads 226 and 228). The writing to a screen is controllable by this.

[0027] The still more detailed flow of interprocess communication is shown in drawing 3 . Drawing 3 is divided into three sections which express the address space of an application task, a personality neutral server, and the main personality server, respectively. These three sections are what [only] was used as an example. The same communication path is used for other servers.

[0028] The application program it runs in the application address space 310 may write data in the window on an indicating-equipment screen. In this case, a demand is performed according to the operating system protocol known by application. An interface intercepts a write request and the decision of processing is made. It inspects whether as for an interface, whether an indicating-equipment page's being in application address space memory and a task can access the page first. When these two conditions are ready, an interface can be directly written in a display. When not ready, a memory page fault occurs.

[0029] Generating of a memory page fault sends a message to the indicating-equipment driver of the personality neutral service address space 312. A device driver tries a required page with the method of assignment in application memory. It inspects whether it is ***** to the main personality server for whether a device driver can permit access to a page first to a requestor-side process, or access of the requestor-side process to a page should be refused or a requestor-side process should wait for a page, and decision. Shortly after a device driver gets to know a situation, it can be permitted, refused or delayed in a demand, without sending a message to the main server. A device driver answers an application process directly.

[0030] When decision is required, a message is sent to the main personality server address space 314. The main personality applies the regulation for permitting or refusing access, and answers a device driver. Next, a device driver answers application.

[0031] A similar process is used also while application is using operating system environments other than the main personality. In that case, the service request which cannot be processed in interface logic is sent to a vice-personality server task for treatment. The vice-personality server 214 must send a message to the main personality server 210, when it is necessary to apply the policy of a system. The main personality server 210 cancels all resource contention.

[0032] The system of this invention hardly needs the excessive overhead for supporting the operating system environment where it operates in parallel, on the occasion of actuation. In a single operating system or a single Mach single personality embodiment, there is usually a service request from application to system level or a kernel. Although it can route in order for Mach to process a message directly or to make a server process it, Mach architecture does not make most overheads increase to this transaction. The embodiment of the desirable message base of an example is based on the model of the minimum message traffic. The application which uses the main personality comes to recognize the same response as being obtained in the single operating system embodiment. A great portion of vice-personality demand is processed promptly similarly. Message traffic increases [a resource contention exception] by inspection of the main personality server 210.

[0033] Therefore, this invention attains the object of performing processing which was able to take adjustment for [two or more] operating systems, and, moreover, attains it by the minimum overhead.

[0034] A system administrator or system installation personnel performs the main personality's selection. Each available operating system is supplied in the main format and a subformat. It chooses which operating system a manager makes the main personality, and all other operating systems are offered as a vice-personality. A vice-personality needs to care about that it is not the slave of the main server. A vice-personality expresses a perfect operating system environment. However, a certain kind of function is taken by the main personality.

[0035] It gets with that of the following and invention is indicated.

[0036] (1) A means to perform basic kernel service, and a main personality means to control the whole system behavior by offering operating system service according to the 1st operating system protocol, An application interface means to receive an operating system service request from an application program,

A vice-personality means to offer operating system service according to one or more different operating system protocols from said 1st operating system protocol, Said means to perform said basic kernel service, and said main personality means, The system equipped with an interface means to link said application interface means and said vice-personality means for supporting the operating system with which plurality lives together.

[0037] (2) A system given in (1) characterized by said means to perform said basic kernel service being the microkernel of the Mach base.

[0038] (3) A system given in (1) to which said interface means evaluates an object-oriented message, and is characterized by including a means to act on them, and a means to evaluate a procedure call and to act on them.

[0039] (4) A system given in (1) characterized by said main personality means being AIX (trademark of U.S. IBM) operating environment.

[0040] (5) A system given in (1) characterized by said main personality means being OS/2 (trademark of U.S. IBM) operating environment.

[0041] (6) A system given in (4) to which said vice-personality means is characterized by including OS/2 operating environment, DOS operating environment, and the Macintosh operating environment.

[0042] (7) A system given in (5) to which said vice-personality means is characterized by including AIX operating environment, DOS operating environment, and the Macintosh operating environment.

[0043] (8) Which of two or more application tasks can be accessed, respectively. Interact with the processor system which has a computer processor and memory. The step which is the approach of operating two or more operating system environments in parallel, and boots the 1st operating system as a main movement environmental process, The step which loads two or more suboperating system environments as a separate process, The step which answers a message from said main movement environment and said two or more suboperating system environments, and loads one or more personality neutral servers, The approach characterized by having the step which performs application as an application process, and the step which uses kernel service and processes an interprocess communication demand and a demand peculiar to a processor system.

[0044] (9) An approach given in (8) characterized by said kernel service being Mach microkernel service of the U.S. Carnegie Mellon University development.

[0045] (10) It is the computer program product which has the computer read possible medium which performs two or more operating systems in parallel, and by which computer program logic was recorded on it. A means to perform basic kernel service, and a main personality means to control the whole system behavior by offering operating system service according to the 1st operating system protocol, An application interface means to receive an operating system service request from an application program, A vice-personality means to offer operating system service according to one or more different operating system protocols from said 1st operating system protocol, The computer program characterized by having an interface means to link said means to perform basic kernel service, said main personality means, said application interface means, and said vice-personality means.

[0046] (11) A computer program given in (10) characterized by said means to perform said basic kernel service being the microkernel of the Mach base.

[0047] (12) A computer program given in (10) to which said interface means evaluates an object-oriented message, and is characterized by including a means to act on them, and a means to evaluate a procedure call and to act on them.

[0048] (13) A computer program given in (10) characterized by said main personality means being AIX operating environment.

[0049] (14) A computer program given in (10) characterized by said main personality means being OS/2 operating environment.

[0050] (15) A computer program given in (13) to which said vice-personality means is characterized by including OS/2 operating environment, DOS operating environment, and the Macintosh operating

environment.

[0051] (16) A computer program given in (14) to which said vice-personality means is characterized by including AIX operating environment, DOS operating environment, and the Macintosh operating environment.

[0052]

[Effect of the Invention] Before writing a system user to DOS, Windows, OS/2, AIX, or other operating systems, while he uses the application of a from succeedingly by offering two or more concurrency operating system personalities, it becomes possible to shift to a new operating system model. Moreover, according to this structure, a developer develops a new operating system and it becomes still easier to make new hardware suit. Especially, the capitalization to the existing operating system service is protected.

[Translation done.]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.**** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the block diagram of the computer system which can carry out the desirable example of this invention.

[Drawing 2] It is the block diagram of the system by the desirable example of this invention.

[Drawing 3] It is the flow chart which shows the process step contained in application use of the desirable example of this invention.

[Description of Notations]

102 CPU

103 System Bus

104 Storage

106 Random Access Memory (RAM)

108 Adapter

110 Input/output Control Unit

112 Display

114 Keyboard

116 Pointing Equipment

202 Basic Microkernel Layer

204 Hardware Layer

206 Operating System Personality

209 Interface

210 The Main Personality Server

211 Interface

212 Personality Neutral Service

213 Interface

214 Vice-Personality Server

215 Interface

310 Application Address Space

312 Personality Neutral Service Address Space

314 The Main Personality Address Space

[Translation done.]